PRÁCTICA 3: Seguridad perimetral – Firewall y NAT

Índice

1-LISTAS DE CONTROL DE ACCESOS (ACL):

Una ACL es una **lista secuencial de comandos** permit o deny, que se aplican (o no) a los paquetes que atraviesan el router en base a información de las cabeceras de capa (capas 3 y 4).

La configuración de la secuencia de comandos es CLAVE.

1.1- Funcionamiento de las ACLs

• Propósito de las ACLs:

1. Limitar el tráfico para mejorar el rendimiento de la red.
2. Controlar el flujo de tráfico.
3. Dar seguridad básica.
4. Filtrar tráfico en base a múltiples criterios:
    - Clasificar el tráfico para ser procesado en VoIP
    - Filtrar el resultado de un "debug"
    - Definir el tráfico que va a ser traducido por NAT…
5. Controlar el acceso a servicios

Funcionan a nivel de IP. Se usan en SVIs y en puertos enrutados.

USO BÁSICO:

3 y 4 del paqueteComandos:

-permit: permite el tráfico

-deny: deniega el tráfico

Permiten convertir un router en un firewall.

Son una lista secuencial de comandos. El orden en que se escriban los comandos es muy importante. El router revisa los comandos por orden. Los comandos que se van añadiendo se añaden al final, y no se puede cambiar el orden de los anteriores. No se puede escribir por el medio de los comandos anteriores, para ello habría que borrar toda la configuración de esa ACL y volver a configurarla entera.

¡¡¡¡ Siempre que bloqueamos el tráfico, tiene que ser hacia ambos sentidos. Hay bloquear también el tráfico de retorno !!!!

- Repaso del establecimiento, mantenimiento y finalización de una conexión TCP

TCP SYN = 1

ACK = 0 -> CONEXIÓN NO ESTABLECIDA

EN EL RESTO DE CASOS -> CONEXIÓN ESTABLECIDA

- Funcionamiento del filtrado estático de paquetes:

– El router controla el acceso entre redes analizando los paquetes entrantes y/o salientes, permitiéndolos o denegándolos, en base a determinados criterios, como:

  • IP origen y destino
  • Campo protocolo
  • Datos específicos de los protocolos empaquetados en IP

– Puertos TCP origen y destino

– Puertos UDP origen y destino

– Comandos ICMP

– El router analiza cada paquete individualmente, enviando (permit) o descartando (deny) cada uno en base a los criterios de coincidencia especificados en cada entrada de la ACL (Access Control Entry, ACE) o sentencia de la ACL

– Cada una de las sentencias es evaluada secuencialmente. Cuando se produce una coincidencia con una de las sentencias, se ejecuta la acción asociada (permit o deny). Si no se cumple la condición, se examina la sentencia siguiente

– Si no se produce coincidencia con ningún conjunto de condiciones Deny any

- Tipos de ACLs:

Hay dos tipos: estándar y extendidas:

-Estándar: filtran por IP origen.

Ejemplo: bloquea todo lo que venga de 192.168.1.10:

access-list 1 deny 192.168.1.10

-Extendidas: filtran por IP origen y destino, puertos y protocolos.

Ejemplo: permitir el tráfico http (puerto 80) desde 192.168.1.0 hacia 172.16.0.0

access-list 101 permit tcp 192.168.1.0 0.0.0.255 172.16.0.0 0.0.0.255 eq 80

Las ACLs estándar y extendidas se pueden crear tanto "numeradas" como "nombradas". Esto quiere decir que es solamente como se identifican:

- ACLs numeradas:

  Se identifican por un número:

  -ACL Estándar: Filtran únicamente por dirección IP origen (rangos 1-99, 1300-1999)

  -ACL Extendida: Filtran por IP origen/destino, protocolo, puertos, etc. (rangos 100-199, 2000-2699)

- ACL Nombradas: utilizan nombres en lugar de números para identificarlas:

  – El nombre puede contener letras y números

  – Se recomienda utilizar letras mayúsculas para el nombre de la ACL

  – El nombre no puede contener ni espacios ni signos de puntuación

  – Se pueden añadir y eliminar entradas de la AC

  Por defecto, el router no tiene ACLs configuradas -> No filtra el tráfico

  – El tráfico que entra en el router se envía hacia el destino en base al contenido de la tabla de enrutamiento

  Al aplicar una ACL a una interfaz, el router debe evaluar si el paquete debe atravesar la interfaz a la que está asignada la ACL o no.

1.2- Máscaras Wildcard

Una máscara wildcard es una secuencia de 32 bits (como una dirección IP) que le dice al router qué bits de una dirección IP debe examinar y cuáles ignorar cuando evalúa una regla de ACL.

Funcionamiento básico

- Bit 0: "Comprobar este bit" (debe coincidir)

- Bit 1: "Ignorar este bit" (puede ser cualquier valor)

Ejemplos:

1- Wildcard: 0.0.0.0

- Significa: "compara toda la IP".
- Útil para una sola IP exacta.

```
access-list 10 permit 192.168.1.10 0.0.0.0
```

2- Wildcard: 0.0.0.255

- Significa: "compara los primeros 3 octetos, ignora el último".
- Se usa para una red entera de 256 direcciones.

access-list 10 permit 192.168.1.0 0.0.0.255

Permite toda la red 192.168.1.0/24 (desde .0 hasta .255).

3- Wildcard: 0.0.3.255

- Compara todo menos los últimos 10 bits.
- Cubre un rango de IPs más grande (por ejemplo, 1024 direcciones).

access-list 10 permit 192.168.0.0 0.0.3.255

Permite desde 192.168.0.0 hasta 192.168.3.255.

Wildcard especial: any y host

- Cisco te da palabras clave para facilitar:

Palabra	Equivale a...	Significa
any	0.0.0.0 255.255.255.255	Cualquier IP
host	IP 0.0.0.0	IP exacta

Ejemplo con any:

access-list 10 permit any # Permite todo el tráfico.

Ejemplo con host:

access-list 10 permit host 192.168.1.10 # Permite solo la IP 192.168.1.10.

Ejemplos a partir de IPs:

192.168.1.0/24 -> 0.0.0.255 (comparar los 24 primeros bits)

192.168.1.100/32 -> 0.0.0.0 (comparar toda la IP)

192.168.1.0/24 La mitad inferior permit -> 0.0.0.127

superior deny

1.3- Guía para la creación y ubicación de ACLs

- Pasos recomendados para la creación de una ACL:

1. Basarse en la política de seguridad
    - Antes de crear una ACL, debes saber qué tráfico está permitido o bloqueado.
2. Describir lo que debe hacer la ACL
    - Es útil hacer una lista escrita: “Permitir esto, bloquear aquello…”
3. Usar un editor de texto
    - Es mejor escribir la ACL fuera del router (por ejemplo en el Bloc de notas), para evitar errores.
4. Probar en un entorno de laboratorio
    - Así te aseguras de que no vas a cortar servicios por error en la red real.
5. Tener un plan de recuperación
    - Por si algo sale mal, debes tener un respaldo o forma rápida de volver atrás.

- Guías para la ubicación de una ACL:

Ubicar bien una ACL en la red es clave para su eficiencia y funcionamiento correcto.

¿Dónde se colocan?

-En routers frontera, entre redes internas y externas.

-Entre zonas de seguridad diferentes (por ejemplo, interna y DMZ).

-Pueden aplicarse en entrada (in) o salida (out) de una interfaz.

Reglas RECOMENDADAS:

1. ACL Estándar → Cerca del DESTINO

- Como solo filtra por IP origen, si la pones cerca del origen podrías bloquear demasiado tráfico.

```
access-list 1 deny 192.168.1.10
interface g0/1
ip access-group 1 in
```

Mejor ponerla en la interfaz que **recibe** tráfico del destino.

2. ACL Extendida → Cerca del ORIGEN

- Puedes filtrar de forma específica (IP, puerto, protocolo…), así que mejor bloquear el tráfico lo antes posible.

Ejemplo:

access-list 101 deny tcp 192.168.1.10 any eq 80

interface g0/0

ip access-group 101 in

Resumen rápido

Tipo de ACL	**¿Dónde ponerla?**	**¿Por qué?**
Estándar	Cerca del **destino**	Solo filtra por IP de origen
Extendida	Cerca del **origen**	Filtra por IP, puertos, protocolo

Explicación de IN vs OUT

- IN: Filtra los paquetes cuando entran a una interfaz del router. Es como un guardia en la puerta de entrada.

- OUT: Filtra los paquetes cuando están a punto de salir por una interfaz del router. Es como un guardia en la puerta de salida.

Filtrar orígenes → in en la interfaz por donde entra el origen.

Filtrar destinos → out en la interfaz por donde sale el paquete al destino.

Fijarse bien en la interfaz a la que hay que asignarle la ACl para saber si es IN o OUT.

IMPORTANTE:

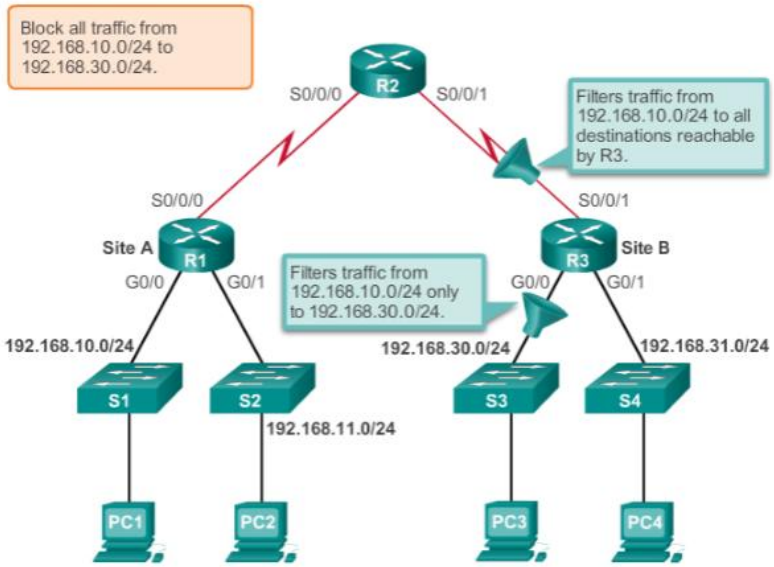
Los paquetes se deniegan automáticamente. Si ponemos un deny, todo lo que no coincida con esta regla será bloqueado automáticamente. Por tanto, si no ponemos un permit general luego, no podremos tener nada de tráfico en esa interfaz.

deny any / deny ip any any implícitos

- Configuraciones de ACL estándar

Escenario1: ejemplo de aplicación de ACL Estándar

- Ejemplo de Aplicación de una ACL Estándar



Este escenario muestra una red corporativa con dos sitios (A y B) conectados a través de un router central (R2). La topología incluye:

- Sitio A (izquierda): Router R1 conectado a dos redes:

    - Red 192.168.10.0/24 (PC1)

    - Red 192.168.11.0/24 (PC2)

- Sitio B (derecha): Router R3 conectado a dos redes:

    - Red 192.168.30.0/24 (PC3)

    - Red 192.168.31.0/24 (PC4)

- Router Central (R2): Conecta ambos sitios mediante enlaces seriales

Objetivo de Seguridad

El objetivo es implementar políticas de seguridad mediante ACLs estándar para controlar el tráfico desde la red 192.168.10.0/24. El objetivo de este escenario es impedir que los usuarios de la red 192.168.10.0/24 (Site A) puedan comunicarse con los usuarios de la red 192.168.30.0/24 (Site B).

Hay diferentes formas de implementar esta restricción, como se muestra en los tres cuadros de texto del diagrama:

1-En R2: Bloquear completamente todo el tráfico que va desde la red 192.168.10.0/24 hacia la red 192.168.30.0/24. Esto es una restricción total.

```
R2(config)# access-list 1 deny 192.168.10.0 0.0.0.255
R2(config)# access-list 1 permit any
R2(config)# interface S0/0/1
R2(config-if)# ip access-group 1 out
```

Bloqueamos solo la ip .30 y permitimos el resto del tráfico.

Se aplica la ACL en dirección OUT en la interfaz que va hacia R3. Esto significa que el filtrado ocurre cuando los paquetes salen de R2 hacia R3.

Se usa OUT porque:

- La ACL se aplica en la interfaz S0/0/1 de R2 que va hacia R3

- Queremos filtrar el tráfico justo antes de que salga de R2 hacia R3

- Al usar OUT, estamos diciendo: "revisa todos los paquetes que están a punto de salir por esta interfaz y bloquea los que vienen de 192.168.10.0/24"

2-En R3: Hay dos situaciones diferentes:

- Opción restrictiva: Filtrar todo el tráfico desde 192.168.10.0/24 hacia cualquier destino alcanzable por R3

```
R3(config)# access-list 2 deny 192.168.10.0 0.0.0.255
R3(config)# access-list 2 permit any
R3(config)# interface S0/0/1
R3(config-if)# ip access-group 2 in
```

Se aplica la ACL en dirección IN en la interfaz que viene de R2. Esto filtra los paquetes cuando entran a R3, bloqueando todo el tráfico de la red 192.168.10.0/24 hacia cualquier destino.

Se usa IN porque:

- La ACL se aplica en la interfaz S0/0/1 de R3 que viene de R2

- Queremos filtrar el tráfico justo cuando entra a R3 desde R2

- Al usar IN, estamos diciendo: "revisa todos los paquetes que están entrando por esta interfaz y bloquea los que vienen de 192.168.10.0/24"

- Opción específica: Filtrar tráfico desde 192.168.10.0/24 específicamente hacia 192.168.30.0/24

```
R3(config)# access-list 3 deny 192.168.10.0 0.0.0.255
R3(config)# access-list 3 permit any
R3(config)# interface G0/0
R3(config-if)# ip access-group 3 out
```

Se aplica la ACL en dirección OUT en la interfaz G0/0 que va hacia la red 192.168.30.0/24. Esto filtra los paquetes justo antes de que salgan hacia la red destino.

La mejor opción es el Caso 3: Opción específica

Al ser una ACL estándar la mejor opción es siempre ponerla en el destino como vimos antes.

1. Ubicación óptima: Las ACLs estándar deben colocarse lo más cerca posible del destino. En este caso, se aplica justo en la interfaz que conecta con la red destino (192.168.30.0/24).

2. Menor impacto: Solo bloquea el tráfico específico hacia 192.168.30.0/24, permitiendo que el tráfico de 192.168.10.0/24 llegue a otras redes (como 192.168.31.0/24).

3. Principio de ACL estándar: Las ACLs estándar solo filtran por dirección de origen, por lo que aplicarlas cerca del destino es más eficiente y evita bloquear tráfico legítimo.

4. Dirección OUT apropiada: Al aplicar la ACL en dirección OUT en la interfaz G0/0, se asegura que el filtrado ocurra justo antes de que los paquetes lleguen a su destino, permitiendo que el router procese normalmente el resto del tráfico.

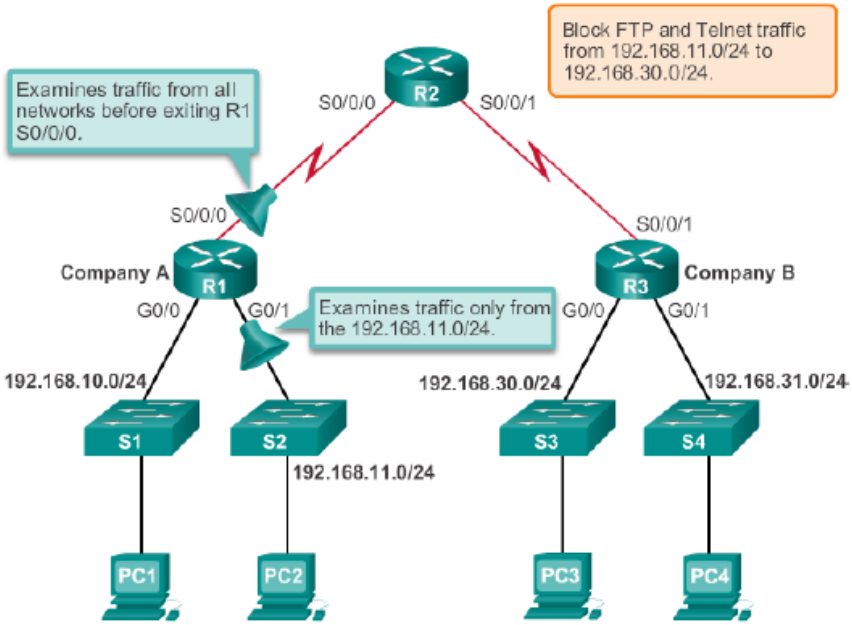
Este enfoque sigue la regla general para ACLs estándar: "Las ACLs estándar deben colocarse lo más cerca posible del destino", ya que solo pueden filtrar por dirección de origen y no por destino.

*Solo usamos comandos de estándar.

- Configuraciones de ACL extendidas

Escenario2: ejemplo de aplicación de ACL Extendida

Ejemplo de Aplicación de una ACL Extendida



Mismo escenario que el anterior.

Objetivo de Seguridad

El objetivo de este escenario es controlar tipos específicos de tráfico entre redes, permitiendo un filtrado más granular que las ACLs estándar. En particular, se busca bloquear servicios específicos (FTP y Telnet) desde la red origen 192.168.11.0/24 hacia una red destino 192.168.30.0/24, manteniendo otros servicios funcionales.

Hay diferentes formas de implementar esta restricción, como se muestra en los tres cuadros de texto del diagrama:

1. En R1: Examinar todo el tráfico de todas las redes antes de que salga por la interfaz S0/0/0 de R1. Examina todo el tráfico

```
R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.30.0 0.0.0.255 eq 21

R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.30.0 0.0.0.255 eq 23

R1(config)# access-list 101 permit ip any any

R1(config)# interface S0/0/0

R1(config-if)# ip access-group 101 out
```

Estos comandos crean una ACL que bloquea FTP (puerto 21) y Telnet (puerto 23) desde la red 192.168.11.0/24 hacia 192.168.30.0/24

Se aplica en dirección OUT en la interfaz S0/0/0, lo que significa que el filtrado ocurre cuando los paquetes están a punto de salir de R1 hacia R2

La dirección OUT aquí significa "revisa los paquetes justo antes de que salgan por esta interfaz"

2-En R2: Bloquear específicamente el tráfico FTP y Telnet desde 192.168.11.0/24 hacia 192.168.30.0/24. Bloquea servicios específicos.

```
R2(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.30.0 0.0.0.255 eq 21

R2(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.30.0 0.0.0.255 eq 23

R2(config)# access-list 101 permit ip any any

R2(config)# interface S0/0/1

R2(config-if)# ip access-group 101 out
```

Esta ACL bloquea exactamente los mismos servicios que en el Caso 1

Se aplica en dirección OUT en la interfaz S0/0/1 de R2, que es la que va hacia R3

La dirección OUT** aquí significa "filtra los paquetes cuando están saliendo hacia R3"

3-En R1: Examinar solo el tráfico que proviene de la red 192.168.11.0/24. Examina solo tráfico específico.

```
R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.30.0 0.0.0.255 eq 21
R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255 192.168.30.0 0.0.0.255 eq 23
R1(config)# access-list 101 permit ip any any
R1(config)# interface G0/1
R1(config-if)# ip access-group 101 in
```

Esta ACL bloquea FTP y Telnet desde 192.168.11.0/24 hacia 192.168.30.0/24

Se aplica en dirección **IN** en la interfaz G0/1, que es la que conecta con la red 192.168.11.0/24

La dirección **IN** aquí significa "revisa los paquetes justo cuando entran a esta interfaz desde la red 192.168.11.0/24"

La mejor opción es el Caso 3 por las siguientes razones:

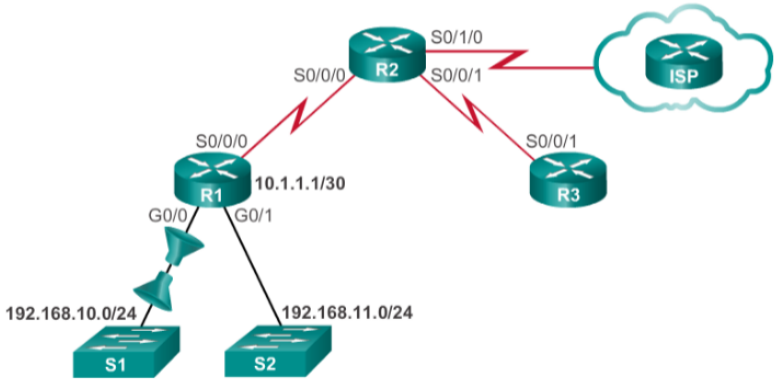
1. **Ubicación óptima**: Las ACLs extendidas deben colocarse lo más cerca posible del origen. En este caso, se aplica justo en la interfaz que conecta con la red origen (192.168.11.0/24).
2. **Eficiencia de recursos**: Al filtrar el tráfico no deseado lo antes posible, se evita que consuma ancho de banda en enlaces intermedios.
3. **Dirección IN apropiada**: Al aplicar la ACL en dirección IN en la interfaz G0/1, se asegura que el filtrado ocurra tan pronto como los paquetes entran al router, antes de que se procesen para enrutamiento.
4. **Principio de ACL extendida**: Las ACLs extendidas deben colocarse lo más cerca posible del origen, ya que pueden filtrar por dirección de origen, destino, protocolo y puerto.

Este enfoque sigue la regla general para ACLs extendidas: "Las ACLs extendidas deben colocarse lo más cerca posible del origen", maximizando la eficiencia de la red y minimizando el procesamiento innecesario de paquetes que eventualmente serían descartados.

Escenario3: ejemplo de aplicación de ACL Extendida II

Ejemplo I: Permitir a los usuarios de la red 192.168.10.0/24 navegar tanto con http como con https.

- La ACL 103 se aplica en la int g0/0 en sentido entrante
- El tráfico de entrada en la LAN 192.168.10.0 solamente se permite si es tráfico de conexiones TCP previamente establecidas




```
R1(config)#access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config)#access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)#access-list 104 permit tcp any 192.168.10.0 0.0.0.255 established
R1(config)#interface g0/0
R1(config-if)#ip access-group 103 in
R1(config-if)#ip access-group 104 out
```

1. access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80

Permite que la red 192.168.10.0/24 pueda iniciar conexiones TCP a cualquier destino usando el puerto 80 (HTTP).

2. access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443

Permite que la red 192.168.10.0/24 pueda iniciar conexiones TCP a cualquier destino usando el puerto 443 (HTTPS).

3. access-list 104 permit tcp any 192.168.0.0 0.0.0.255 established

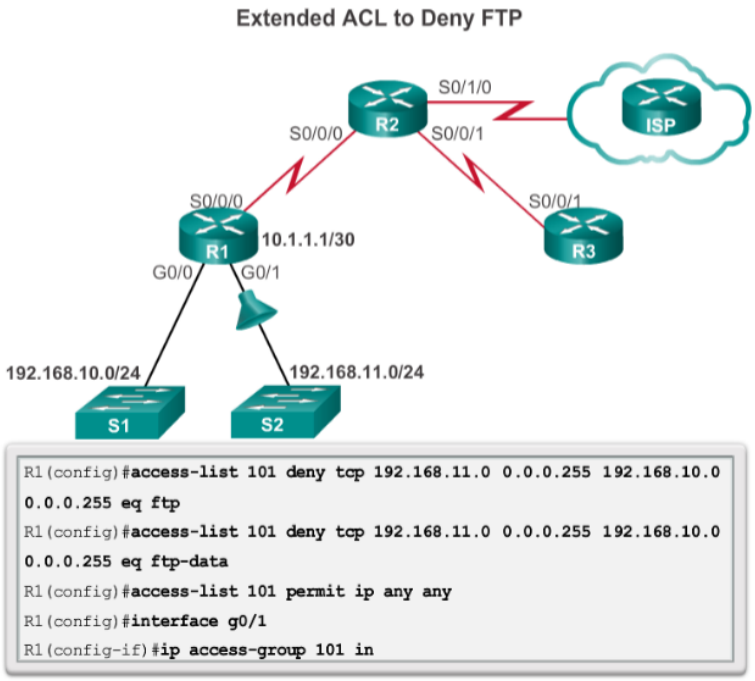
Permite que cualquier host pueda enviar tráfico TCP hacia la red 192.168.10.0/24, pero solo si la conexión ya fue establecida previamente (es decir, si es una respuesta a una conexión iniciada desde la LAN).

4. interface g0/0
5. ip access-group 103 in
6. ip access-group 104 out

Básicamente la regla 103 permite que el tráfico pueda salir (in->porque sale hacia el router) hacia los puertos 80 y 443 y la regla 104 permite que las respuestas de Internet puedan volver a los usuarios, pero solo si iniciaron la conexión (out -> porque ese tráfico (las respuestas) están saliendo del router hacia la LAN (G0/0)

Escenario3: ejemplo de aplicación de ACL Extendida III

Ejemplo II: En este ejemplo se deniega el tráfico FTP desde la subred 192.168.11.0 a la 192.168.10.0, pero se permite el resto del tráfico



Escenario4: ejemplo de aplicación de ACL Extendida Nombrada

Mismo caso y objetivo que el anterior pero con ACL extendida nombrada

El ORDEN de las sentencias en las ACLs es muy FUNDAMENTAL Veamos un ejemplo:

access-list 2 deny host 192.168.10.10

access-list 2 permit 192.168.10.0 0.0.0.255

access-list 2 deny 192.168.0.0 0.0.255.255

access-list 2 permit 192.0.0.0 0.255.255.255

Pregunta 1:

Un paquete llega al router desde la IP de origen 192.168.10.10. ¿Será permitido o bloqueado por la ACL?

Bloqueado (primera línea)

Pregunta 2:

Un paquete llega desde 192.168.10.25. ¿Será permitido o bloqueado por la ACL?

Permitido (segunda línea)

Pregunta 3:

¿Y si llega un paquete desde 192.168.20.1? ¿Será permitido o bloqueado?

Bloqueado (tercera línea)

Pregunta 4:

¿Qué pasa si el paquete viene de 192.200.1.1? ¿Permitido o bloqueado?

Permitido (cuarta línea)

1.4- Modificación de las ACL

1-Editor de textos externos

-Mostrar la configuración de la ACL actual: show running-config | include access-list [num ACL]

Salida:

access-list 1 deny host 192.168.10.99

access-list 1 permit 192.168.0.0 0.0.0.255

-Copiar esta configuración anterior en un editor de texto

-Eliminar la ACL existente: no access-list [num ACL]

Ejemplo: no access-list 1

-Recrear la ACL completa con los cambios deseados

access-list 1 deny host 192.168.10.10

access-list 1 permit 192.168.0.0 0.0.0.255

-Verificar los cambios: show running-config | include access-list [num ACL]

2-Números de secuencia:

Para cada ACL que hemos creado según el orden de los comandos que hemos introducido se le ha asignado un número de secuencia de menos a más a las instrucciones que hemos ido añadiendo a la ACL (de la primera-menor número a la más reciente-mayor número)

- Reemplazar una instrucción por otra:

  -Ver números de secuencia: show access-lists [num ACL]

  Ejemplo:

  Standart IP access list 1

  10 deny 192.168.10.99

  20 permit 192.168.0.0, wildcard bits 0.0.255.255

  -Acceder a la lista de números de secuencia, eliminar un número de secuencia y volverlo a asignar:

  Ejemplo: vamos a borrar el número de secuencia 10 y volver a asignarlo con otros datos

conf t

ip access-list standard 1

no 10

10 deny host 192.168.10.10

end

-Verificar los cambios: show access-lists

show access-list [num ACL]

- Inserción de una nueva instrucción (ACE – Access Control Entry):

Añadir una nueva entrada en una ACL. Mismo caso que el anterior, pero sin eliminar nada y añadiendo nuevas instrucciones de por medio.

-Ver números de secuencia: show access-lists [num ACL]

- Acceder a la lista de números de secuencia, y añadir una nueva instrucción de por medio

conf t

ip access-list standard [num ACL]

[número-secuencia] {permit|deny} [criterios]

end

-Verificar los cambios: show access-lists

show access-list [num ACL]

3-Verificación de ACLs:

La verificación de ACLs consiste en comprobar que la ACL está correctamente aplicada a una interfaz y está funcionando como esperas (es decir, que realmente está filtrando tráfico).

Esto incluye:

-Ver si la ACL está aplicada en la interfaz correcta.

-Saber si está aplicada en modo in o out.

-Ver si está bloqueando o permitiendo tráfico (usando contadores).

show ip interface [interface]: Este comando te permite ver el **estado de una interfaz**, y si tiene alguna ACL aplicada

Ejemplo:
show ip interface GigabitEthernet0/1

Salida:
GigabitEthernet0/1 is up, line protocol is up

Internet address is 192.168.20.1/24

Inbound access list is 10

Outgoing access list is not set

¿Qué significa eso?

-Inbound access list is 10 → La ACL número 10 está aplicada en modo in (tráfico que entra a través de esta interfaz).

-Outgoing access list is not set → No hay ninguna ACL en modo out en esta interfaz.

4-Visualización de la Información Estadística:

Que puedes ver cuántos paquetes han sido permitidos o denegados por cada regla de la ACL.

Para esto hay que usar el comando:

show access-lists

Salida:
Standard IP access list 10

10 deny 192.168.1.10 (10 matches)

20 permit 192.168.1.0, wildcard bits 0.0.0.255 (52 matches)

10 matches → 10 paquetes han sido denegados por esa línea.

52 matches → 52 paquetes han sido **permitidos** por esa línea.

2-CONTROL DE ACCESO BASADO EN EL CONTEXTO (CBAC):

2.1– Características de las CBAC

- CBAC (Context-Based Access Control) es una solución de seguridad de firewall de capa de aplicación.

– Filtran de forma inteligente paquetes TCP y UDP basándose en información de las sesiones de los protocolos de capa de aplicación

– Proporciona filtrado con estado de capa de aplicación.

– Diseñadas para filtrar tráfico de aplicaciones multimedia y protocolos que requieren de múltiples canales de comunicación tales como FTP, H.323, SIP, …

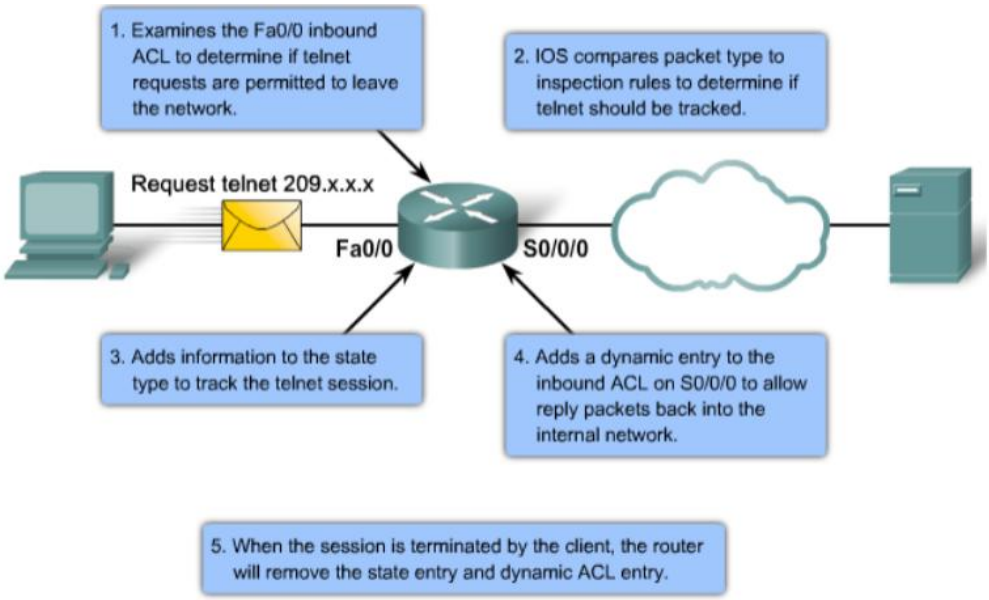
Básicamente CBAC **inspecciona activamente las conexiones** que pasan por el router y permite automáticamente el **tráfico de retorno**, **sin necesidad de ACLs complicadas** con la palabra established.

CBAC mejora a las ACLs tradicionales:

ACL tradicional	**CBAC**
Reglas estáticas	Inspección dinámica del tráfico
Necesitas permitir respuestas manualmente (established)	Las respuestas se permiten automáticamente si la conexión fue iniciada desde adentro
Menos inteligente	Más "stateful" (consciente del estado de conexión)
No detecta protocolos complejos (FTP, VoIP) fácilmente	Puede entender protocolos que usan múltiples puertos (como FTP activo)

2.2– Funcionamiento de las CBAC

Escenario5: Funcionamiento de las CBACs



1. **Paso 1: Packet Inspection (Inspección de Paquetes) -> paquetes de salida**
    - **Descripción**: El router recibe un paquete de datos que entra en la red.
    - **Acción**: Se inspecciona el paquete para determinar si coincide con las reglas de inspección definidas.
2. **Paso 2: Session Creation (Creación de Sesión) -> paquetes de salida**
    - **Descripción**: Si el paquete es permitido, se crea una entrada en la tabla de estado.
    - **Acción**: Se registra la información de la conexión, como direcciones IP y números de puerto.
3. **Paso 3: State Comparison (Comparación de Estado) -> paquetes de retorno**
    - **Descripción**: Cuando un paquete de respuesta llega, se compara con las entradas en la tabla de estado.
    - **Acción**: Si la conexión ya existe, se permite el tráfico de retorno.
4. **Paso 4: Temporary Opening (Apertura Temporal) -> paquetes de retorno**
    - **Descripción**: Si la conexión no existe, se puede crear una apertura temporal en la ACL.
    - **Acción**: Esta apertura permite que el tráfico de respuesta regrese a la red interna.
5. **Paso 5: Session Timeout (Tiempo de espera de sesión)**
    - **Descripción**: Si la conexión no se utiliza durante un tiempo determinado, se elimina de la tabla de estado.

- **Acción**: Esto ayuda a liberar recursos y mantener la seguridad.

Conclusión: La imagen ilustra cómo CBAC gestiona el tráfico de red de manera dinámica, permitiendo conexiones legítimas y asegurando que el tráfico no autorizado sea bloqueado. Este proceso mejora la seguridad de la red al permitir solo el tráfico que ha sido previamente autorizado.

RESUMEN FUNCIONAMIENTO:

1-**Entradas temporales en ACLs**:

- Cuando un dispositivo dentro de la red pide algo (por ejemplo, acceder a una página web), se crea una "entrada temporal" que permite que la **respuesta** de esa solicitud vuelva sin problemas.
- Esta entrada se **elimina automáticamente** cuando la conexión se cierra o después de un tiempo sin actividad.

2- **Inspección flexible**:

- **El administrador** decide qué tipo de tráfico inspeccionar (como **HTTP** o **FTP**) y en qué dirección (entrante o saliente).
- Se puede inspeccionar en **una o dos direcciones** (de dentro hacia fuera y de fuera hacia dentro).

3- **Reglas de inspección**:

- Se crean reglas para inspeccionar el tráfico que sale. Si ese tráfico es **permitido**, la respuesta regresa.
- Si el tráfico no pasa la inspección, **se bloquea**.

4- **Protección contra ataques**:

- CBAC puede limitar la cantidad de **conexiones abiertas** para evitar **ataques DoS** (cuando un atacante intenta saturar el sistema con muchas conexiones).

2.3–Configuración de las CBAC

1. Elegir una interfaz

- **Interfaz interna**: Donde comienza el tráfico (por ejemplo, tu red interna).
- **Interfaz externa**: Donde sale el tráfico hacia internet.
- Puedes aplicar la inspección **en una o ambas direcciones** (entrante y saliente).

2. Configurar la ACL (Lista de Control de Acceso)

- Se crea una ACL para **permitir o denegar** el tráfico que quieres inspeccionar.

3. Definir las reglas de inspección

Se crean reglas de inspección para protocolos específicos como TCP, UDP, ICMP, etc.

Ejemplo de comando para crear una regla de inspección:

ip inspect name NOMBRE_REGLA PROTOCOLO [timeout SEGUNDOS]

*timeout: tiempo que dura la sesión si no hay actividad.

Ejemplo:

ip inspect name FIREWALL tcp timeout 300

ip inspect name FIREWALL udp timeout 30

ip inspect name FIREWALL icmp timeout 10

*Una regla de inspección puede contener varios protocolos al igual que una ACL.

4. Aplicar las reglas de inspección a una interfaz

interface INTERFAZ

ip inspect NOMBRE_REGLA {in|out}

*normalmente se usa in

2.4–Troubleshooting CBACs

Es el proceso para revisar si las CBAC están funcionando correctamente, si están creadas las reglas temporales y si están permitiendo el tráfico esperado.

- show ip inspect [opciones]: permite ver que reglas de inspección están activas, que sesiones están abiertas y que ACLs temporales se han creado automáticamente.

Ejemplos de opciones:

- show ip inspect config: muestra la configuración actual de inspección
- show ip inspect sessions: muestra las conexiones activas que están siendo inspeccionadas
- show ip inspect all: muestra toda la configuración.

*debug ip inspect: habilita la depuración de las funciones de inspección de CBAC en dispositivos Cisco.

3-CONFIGURACIÓN DE NAT (NETWORK ADRESS TRANSLATION)

NAT: es un proceso que permite a dispositivos en una red privada, que utilizan direcciones IP privadas, conectarse a Internet utilizando una dirección IP pública

3.1–Base Teórica

La traducción de direcciones de red (Network Address Translation) consiste en sustituir la dirección IP de origen (habitualmente), la dirección IP de destino (muy rara vez) o ambas cuando una paquete que cumple unas determinadas características es enviado a través de un dispositivo NAT como puede ser un router.

Rango de direcciones IP privadas (RFC 1918):

• Clase A: 10.0.0.0 a 10.255.255.255 (1 red)

• Clase B: 172.16.0.0 a 172.31.255.255 (16 redes)

• Clase C: 192.168.0.0 192.168.255.255 (255 redes

TIPOS DE NAT:

- **NAT Estático**: Asigna una IP pública fija a una IP privada.
- **NAT Dinámico**: Utiliza un grupo de IPs públicas para asignar dinámicamente a las IPs privadas.
- **NAT Sobrecargado (PAT)**: Permite que múltiples IPs privadas compartan una sola IP pública utilizando diferentes números de puerto.
- **Port Mapping / Port Forwarding**: Es una asignación estática de IP + Puerto privados con una IP + Puerto público, lo que permite acceder desde la red pública (Internet) a determinados puertos de la red privada

Escenario Base:

OBJETIVO: Proteger la estructura interna de la red corporativa configurada en las prácticas anteriores

Tabla1: tráfico permitido desde las subredes de usuarios hacia otras subredes (VLAN 10, VLAN 20, VLAN 30 Y VLAN 40) servidores e Internet.

Tráfico Permitido	10.1.1.0 /24	10.1.2.0/ 24	10.1.3.0 /24	10.1.4.0 /24	10.2.1.0 /24	10.2.2.0 /24	10.2.3.0/ 24	10.2.4.0 /24	10.0.0.0/ 8	10.255.1.0/24	10.255.2.024	Internet
10.1.1.0/24	NA	-	-	-	-	-	-	-	-	HTTP a todos los servidores de la Subred 10.255.1.0/24 DNS-UDP a 10.255.1.10	-	HTTP, HTTPS, ICMP
10.1.2.0/24	-	NA	-	-	-	-	-	-	-	-	HTTP a todos los servidores de la Subred 10.255.2.0/24 SMTP, POP3 Y DNS (UDP) A 10.255.2.10	HTTP, HTTPS, ICMP
10.1.3.0/24	-	-	NA	-	-	-	-	-	-	HTTP a todos los servidores de la Subred 10.255.1.0/24 DNS-UDP a 10.255.1.10	-	HTTP, HTTPS, ICMP
10.1.4.0/24	-	-	-	NA	-	-	-	-	-	-	HTTP a todos los servidores de la Subred 10.255.2.0/24 SMTP, POP3 Y DNS (UDP) A 10.255.2.10	HTTP, HTTPS, ICMP
10.2.1.0/24	-	-	-	-	NA	-	-	-	-	HTTP a todos los servidores de la Subred 10.255.1.0/24 DNS-UDP a 10.255.1.10	-	HTTP, HTTPS, ICMP
10.2.2.0/24	-	-	-	-	-	NA	-	-	-	-	HTTP a todos los servidores de la Subred 10.255.2.0/24 SMTP, POP3 Y DNS (UDP) A 10.255.2.10	HTTP, HTTPS, ICMP
10.2.3.0/24	-	-	-	-	-	-	NA	-	-	HTTP a todos los servidores de la Subred 10.255.1.0/24 DNS-UDP a 10.255.1.10	-	HTTP, HTTPS, ICMP
10.2.4.0/24	-	-	-	-	-	-	-	NA	-	-	HTTP a todos los servidores de la Subred 10.255.2.0/24 SMTP, POP3 Y DNS (UDP) A 10.255.2.10	HTTP, HTTPS, ICMP

NA: No aplica. Las ACLs no pueden controlar el tráfico dentro de una LAN, puesto que la comunicación entre dispositivos no se hace a través del firewall
- : No se permite ningún tipo de tráfico
La primera columna representa el origen del tráfico y los encabezamientos del resto de columnas el destino.

Tabla2:

-Tráfico permitido desde los servidores hacia las redes internas e Internet.

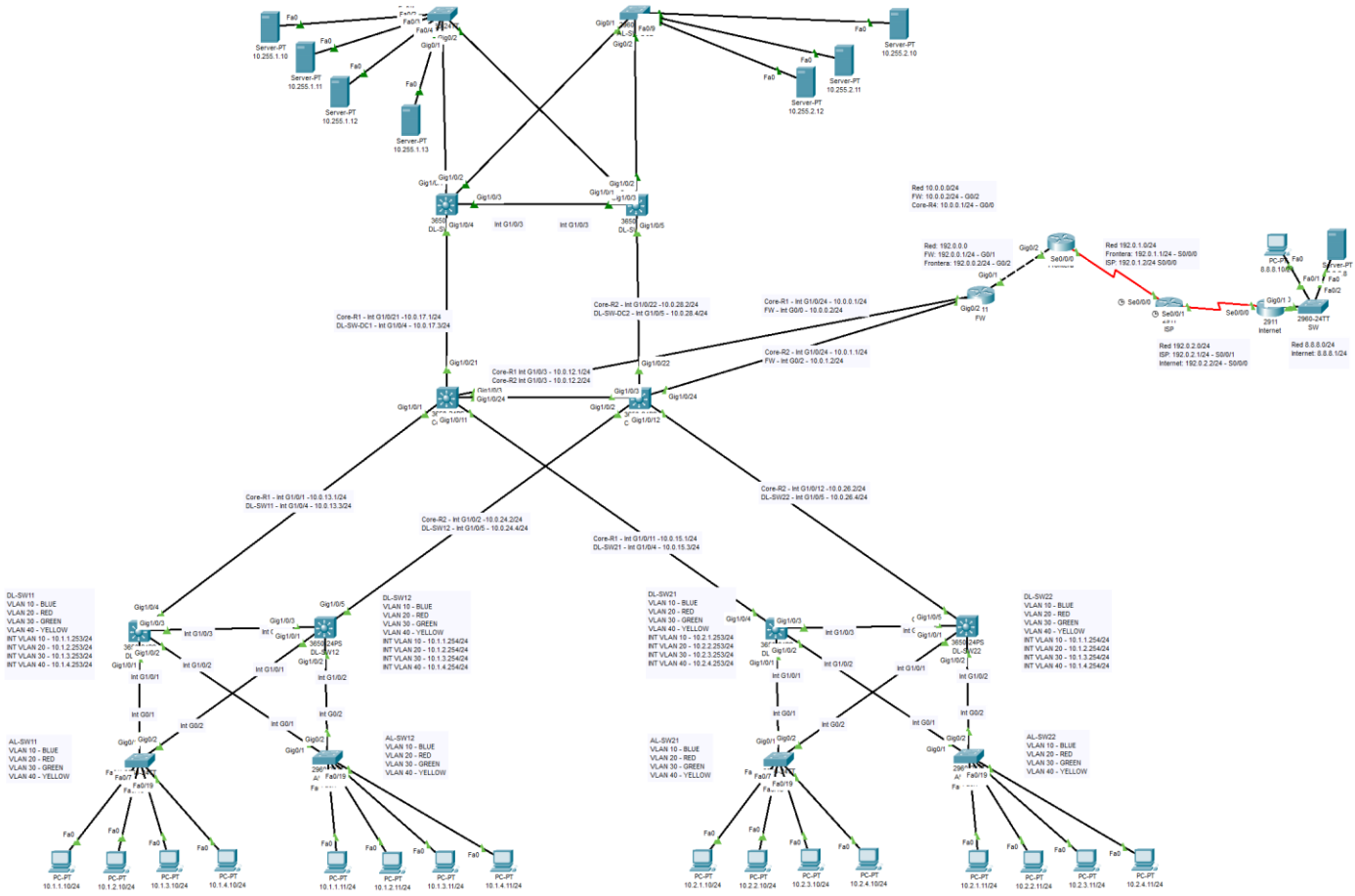
-Tráfico permitido desde Internet hacia las redes internas y servidores

Tráfico Permitido	10.0.0.0/8	10.1.1.0 /24	10.1.2.0/24	10.1.3.0 /24	10.1.4.0 /24	10.2.1.0 /24	10.2.2.0 /24	10.2.3.0/24	10.2.4.0 /24	10.255.1.0/24	10.255.2.024	Internet
10.255.1.0/24	-	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	NA	-	HTTPS, ICMP
10.255.2.0/24	-	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	-	NA	HTTPS, ICMP
Internet	-	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	Tráfico de respuesta a peticiones procedentes de la subred	HTTPS a 10.255.1.10 y tráfico de retorno	HTTPS a 10.255.2.10 y tráfico de retorno	NA

Subredes: 10.1.x.0/24 y 10.2.x.0/24

Servidores: 10.255.1.0/24 y 10.255.2.0/24

Internet



Tecnologías a utilizar

Para resolver de forma adecuada los desafíos que se plantean en esta práctica es necesario que utilicéis en el escenario las tres tecnologías vistas durante la explicación de esta práctica:

• ACLs estándar: configurada/s en el router frontera para evitar que llegue a Firewall todo el tráfico procedente de Internet cuyas direcciones IP de origen sean sospechosas de estar relacionadas con un ataque (IPs marcianas).

• ACLs extendidas: switches de capa de distribución y Firewall

• CBAC: Firewall

Para resolver de forma adecuada los desafíos que se plantean en esta práctica es necesario que utilicéis en el escenario, dos de las tecnologías de traducción de direcciones vistas durante la explicación de esta práctica:

• Port Forwarding / Mapeo de puertos

• NAT Dinámico con sobrecarga

La especificación de cómo debe llevarse a cabo la traducción de direcciones IP, es la siguiente:

a) Las direcciones privadas 10.1.0.0/16 se deben traducir a 192.0.0.10

b) Las direcciones privadas 10.2.0.0/16 se deben traducir a 192.0.0.11

c) Las direcciones privadas 10.3.0.0/16 se deben traducir a 192.0.0.12

d) Las direcciones privadas 10.255.1.0/24 se deben traducir a 192.0.0.13

e) La IP y Puerto 10.255.1.10:443 se debe publicar como 192.0.0.13:443

f) Las direcciones privadas de 10.255.2.0/24 se deben publicar como 192.0.0.14

g) Las IP y Puerto 10.255.2.10:443 se debe publicar como 192.0.0.14:443

Que hay que hacer:

1. Router Frontera:

    1. ACLs estándar para bloquear IPs marcianas (IPs sospechosas)
    2. Máscara correcta para bloquear direcciones multicast y reservadas (31.255.255.255)

2. Router Firewall:

    1. CBAC configurado correctamente para TCP, UDP e ICMP
    2. ACLs extendidas para controlar el tráfico entrante y saliente
    3. NAT dinámico con sobrecarga (PAT) configurado correctamente para las diferentes redes:
        1. 10.1.0.0/16 → 192.0.0.10
        2. 10.2.0.0/16 → 192.0.0.11
        3. 10.3.0.0/16 → 192.0.0.12
        4. 10.255.1.0/24 → 192.0.0.13
        5. 10.255.2.0/24 → 192.0.0.14
    4. Port Forwarding para los servidores web (puerto 443)

3. Switches de Distribución de Servidores (DL-SW-DC1, DL-SW-DC2):

    1. ACLs extendidas para controlar el tráfico entre VLANs
    2. Reglas para permitir protocolos de enrutamiento (OSPF) y redundancia (HSRP)

4. Switches de Distribución de Redes Internas (DL-SW11, DL-SW12, DL-SW21, DL-SW22):

    1. ACLs para VLANs impares y pares
    2. Configuración específica para cada subred según la política de seguridad
    3. Ahora incluye la configuración completa de DL-SW21 y DL-SW22

La configuración cumple con la política de seguridad especificada en la matriz de envío del enunciado, permitiendo:

- Tráfico HTTP/HTTPS/ICMP desde las redes internas hacia Internet
- Acceso HTTP a los servidores web internos según la política
- Acceso a servicios de correo (SMTP/POP3) y DNS según corresponda
- Tráfico de retorno para las conexiones establecidas
- Protocolos de control y enrutamiento (OSPF, HSRP)

Implementación de las tecnologías:

- ACLs estándar en el router frontera
- ACLs extendidas en los switches de distribución y firewall
- CBAC en el firewall
- NAT dinámico con sobrecarga
- Port Forwarding / Mapeo de puertos

La configuración está lista para la defensa de la práctica.

PRACTICA 3:

license boot module c2900 technology-package securityk9

*ROUTERS INTERNET E ISP: nada que configurar

1-ROUTER FRONTERA: bloquear ips marcianas con ACLs estándar

! Configuración de ACLs estándar para bloquear IPs marcianas

- ip access-list standard [nombre]
- {permit|deny} descripcion [dirección-origen] [wildcard]

ip access-list standard BLOCK_BOGON_IPS

deny 0.0.0.0 0.255.255.255 ¡ Bloquear red 0.0.0.0/8

deny 10.0.0.0 0.255.255.255 ! Bloquea IPs privadas clase A que no deberían venir de Internet

deny 172.16.0.0 0.15.255.255 ! Bloquea IPs privadas clase B que no deberían venir de Internet

deny 192.168.0.0 0.0.255.255 ! Bloquea IPs privadas clase C que no deberían venir de Internet

deny 127.0.0.0 0.255.255.255 ! Bloquea direcciones de loopback

deny 224.0.0.0 31.255.255.255 ! Bloquea direcciones multicast y reservadas

permit any ! Permite todo el resto del tráfico

! Aplicación de la ACL a la interfaz externa

interface Serial0/0/0 ! Interfaz hacia Internet

ip access-group BLOCK_BOGON_IPS in ! Aplica la ACL al tráfico entrante desde Internet

-> Con esto decimos: Quiero bloquear estas IPs si entran desde Internet hacia mi red (para que así no llegue al Firewall)

2-ROUTER FIREWALL

conf t

****
*

2.1 Configuración de CBAC

! Definir la inspección para el tráfico de retorno:

- ip inspect name inspection_name protocol [alert] {on | off}] [audit-trail {on | off}] [timeout seconds]

ip inspect name RETURN-TRAFFIC tcp timeout 3600

ip inspect name RETURN-TRAFFIC udp timeout 30

ip inspect name RETURN-TRAFFIC icmp timeout 10

! Aplicar la inspección CBAC a la interfaz de salida

interface GigabitEthernet0/1

ip inspect RETURN-TRAFFIC out

****
****

2.2 - Configuración de ACLs en el firewall

-ACLS ESTÁNDAR:

! Listas de acceso para las direcciones privadas

access-list 1 permit 10.1.0.0 0.0.255.255 ! Permitir el tráfico de la red 10.1.0.0/16

access-list 2 permit 10.2.0.0 0.0.255.255 ! Permitir el tráfico de la red 10.2.0.0/16

access-list 3 permit 10.3.0.0 0.0.255.255 ! Permitir el tráfico de la red 10.3.0.0/16 (red que se usa en NAT)

access-list 4 permit 10.255.1.0 0.0.0.255 ! Permitir el tráfico de la red 10.255.1.0/24

access-list 5 permit 10.255.2.0 0.0.0.255 ! Permitir el tráfico de la red 10.255.2.0/24

-ACLS EXTENDIDAS:

Vamos a crear 2 ACLs para permitir el tráfico entrante y saliente para que las direcciones traducidas (que traduciremos más abajo con NAT) puedan acceder a Internet. Estas direcciones 192.0.0.x deben ser capaces de enviar y recibir tráfico de Internet).

Queremos que estas direcciones traducidas puedan acceder a HTTP, HTTPS e ICMP

! ACL Extendida para tráfico de salida de las futuras traducciones NAT

(como vamos a traducir ips a

ip access-list extended TRAFFIC-OUT

permit tcp host 192.0.0.13 eq 80 any gt 1023 !Permite tráfico HTTP desde puertos de origen mayores a 1023

permit tcp host 192.0.0.14 eq 80 any gt 1023 !Permite tráfico HTTP desde esa red al puerto 443

permit tcp host 192.0.0.13 eq 443 any gt 1023 !Permite tráfico HTTPS desde puertos de origen mayores a 1023

permit tcp host 192.0.0.14 eq 443 any gt 1023 !Permite tráfico HTTPS desde esa red al puerto 443

permit tcp 192.0.0.0 0.0.0.255 any eq 80 ! Permitir HTTP desde cualquier 192.x.x.x

permit tcp 192.0.0.0 0.0.0.255 any eq 443 ! Permitir HTTPS desde cualquier 192.x.x.x

permit icmp 192.0.0.0 0.0.0.255 any

! ACL Extendida para tráfico de entrada de las futuras traducciones NAT

ip access-list extended TRAFFIC-IN

permit tcp any gt 1023 host 192.0.0.13 eq 80

permit tcp any gt 1023 host 192.0.0.14 eq 80

permit tcp any gt 1023 host 192.0.0.13 eq 443

permit tcp any gt 1023 host 192.0.0.14 eq 443

permit tcp any established any ! Permitir tráfico de retorno establecido

permit icmp any 192.0.0.0 0.0.0.255 echo-reply ! Permitir respuestas ICMP

! Aplicar la ACLs de tráfico de entrada y salida en la interfaz correspondiente

interface GigabitEthernet0/1

ip access-group TRAFFIC-IN in

ip access-group TRAFFIC-OUT out

****

2.3 - Configuración de NAT

! Definir los pools de NAT:

- ip nat pool <nombre> <ip-inicial> <ip-final> netmask <máscara>

ip nat pool NAT-POOL-ED1 192.0.0.10 192.0.0.10 netmask 255.255.255.0

ip nat pool NAT-POOL-ED2 192.0.0.11 192.0.0.11 netmask 255.255.255.0

ip nat pool NAT-POOL-ED3 192.0.0.12 192.0.0.12 netmask 255.255.255.0

ip nat pool NAT-POOL-ED4 192.0.0.13 192.0.0.13 netmask 255.255.255.255

ip nat pool NAT-POOL-ED5 192.0.0.14 192.0.0.14 netmask 255.255.255.255

! Configurar NAT dinámico con sobrecarga (PAT):

- ip nat inside source list <número ACL> interface <tipo-número> overload

Lo que vamos a hacer aquí es que todas las ips privadas de cada subred (en este caso 5 subredes (10.1.x.x, 10.2.x.x, 10.3.x.x, 10.255.1.x, 10.555.2.x) se van a traducir a una ÚNICA IP Pública:

-Cualquier ip de la red 10.1.x.x -> 192.0.0.10

-Cualquier ip de la red 10.2.x.x -> 192.0.0.11

-Cualquier ip de la red 10.3.x.x -> 192.0.0.12

-Cualquier ip de la red 10.255.1.x -> 192.0.0.13

-Cualquier ip de la red 10.255.2.x -> 192.0.0.14

ip nat inside source list 1 pool NAT-POOL-ED1 overload ! Traducción dinámica para la red 10.1.0.0/16

ip nat inside source list 2 pool NAT-POOL-ED2 overload ! Traducción dinámica para la red 10.2.0.0/16

ip nat inside source list 3 pool NAT-POOL-ED3 overload ! Traducción dinámica para la red 10.3.0.0/16

ip nat inside source list 4 pool NAT-POOL-ED4 overload ! Traducción dinámica para la red 10.255.1.0/24

ip nat inside source list 5 pool NAT-POOL-ED5 overload ! Traducción dinámica para la red 10.255.2.0/24

! Port Forwarding para los servidores: configurar NAT estático para los puertos 443 de los servidores

- ip nat inside source static [tcp|udp] <ip-local-interna> <puerto-local-interno> <ip-global-interna> <puerto-global-interno>

ip nat inside source static tcp 10.255.1.10 443 192.0.0.13 443

ip nat inside source static tcp 10.255.2.10 443 192.0.0.14 443

! Configuración de NAT en las interfaces correspondientes

interface GigabitEthernet0/0

 ip nat inside

interface GigabitEthernet0/1

 ip nat outside

interface GigabitEthernet0/2

 ip nat inside

Esto es lo que se hace ahora:

Tráfico entre redes internas:

- Las ACLs en los switches de distribución permiten exactamente el tráfico especificado en la matriz

- Redes impares (10.x.1.0/24, 10.x.3.0/24) pueden acceder a servidores web (10.255.1.0/24)

- Redes pares (10.x.2.0/24, 10.x.4.0/24) pueden acceder a servidores de correo (10.255.2.0/24)

Acceso a Internet:

- Todas las redes internas pueden acceder a HTTP, HTTPS e ICMP hacia Internet

- El tráfico de retorno correspondiente está permitido

- La traducción NAT permite que las IPs privadas accedan a Internet usando IPs públicas

Acceso desde Internet:

- Solo se permite acceso HTTPS a los servidores específicos (10.255.1.10 y 10.255.2.10)

- El port forwarding publica estos servidores como 192.0.0.13:443 y 192.0.0.14:443

Protección perimetral:

- El router frontera bloquea IPs marcianas/bogon

- El firewall implementa CBAC para inspeccionar el tráfico de retorno

- Las ACLs extendidas filtran el tráfico entrante y saliente según la política

---

3-SWITCHES DE DISTRIBUCIÓN de SERVIDORES (DL-SW-DC1, DL-SW-DC2):

DL-SW-DC1:

```
! ACL para VLAN10 - Usuarios Web   -> red 10.255.1.0/24
ip access-list extended CONTROL-VLAN10
  permit ospf any any                          ! Permitir OSPF
  permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985     ! Permitir CDP
  permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985   ! Permitir HSRP
  permit tcp 10.255.1.0 0.0.0.255 eq 443 10.0.1.0 0.255.254.255 gt 1023   ! HTTPS a red interna
  permit tcp 10.255.1.0 0.0.0.255 eq www 10.0.1.0 0.255.254.255 gt 1023   ! HTTP a red interna
  permit udp host 10.255.1.10 eq domain 10.0.1.0 0.255.254.255 gt 1023   ! DNS desde servidor
  deny ip any 10.0.0.0 0.255.255.255                ! Denegar acceso a red interna
  permit tcp host 10.255.1.10 eq 443 any             ! Permitir HTTPS desde servidor
  permit tcp 10.255.1.0 0.0.0.255 any eq 443          ! Permitir HTTPS general
  permit icmp 10.255.1.0 0.0.0.255 any               ! Permitir ICMP (ping)

! ACL para VLAN20 - Usuarios Mail   -> red 10.255.2.0/24
ip access-list extended CONTROL-VLAN20
  permit ospf any any                          ! Permitir OSPF
  permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985     ! CDP
  permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985   ! HSRP
  permit tcp 10.255.2.0 0.0.0.255 eq 443 10.0.0.0 0.255.254.255 gt 1023   ! HTTPS a red interna
  permit tcp 10.255.2.0 0.0.0.255 eq www 10.0.0.0 0.255.254.255 gt 1023   ! HTTP a red interna
  permit tcp host 10.255.2.10 eq pop3 10.0.0.0 0.255.254.255 gt 1023     ! POP3 desde servidor
  permit tcp host 10.255.2.10 eq smtp 10.0.0.0 0.255.254.255 gt 1023     ! SMTP desde servidor
  permit udp host 10.255.2.10 eq domain 10.0.0.0 0.255.254.255 gt 1023   ! DNS desde servidor
  deny ip any 10.0.0.0 0.255.255.255                ! Denegar acceso general a red interna
  permit tcp host 10.255.2.10 eq 443 any             ! HTTPS desde servidor
  permit tcp 10.255.2.0 0.0.0.255 any eq 443          ! HTTPS general
```

```
 permit icmp 10.255.2.0 0.0.0.255 any              ! ICMP (ping)


!Aplicación de ACLs en interfaces VLAN
interface Vlan10
 ip access-group CONTROL-VLAN10 in              ! Aplicar ACL en entrada

interface Vlan20
 ip access-group CONTROL-VLAN20 in              ! Aplicar ACL en entrada
```

DL-SW-DC2:

```
ip access-list extended CONTROL-VLAN10                  ! ACL para restringir tráfico de la VLAN10
 permit ospf any any                            ! Permitir OSPF
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985      ! Permitir CDP
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985    ! Permitir HSRP
 permit tcp 10.255.1.0 0.0.0.255 eq 443 10.0.1.0 0.255.254.255 gt 1023  ! HTTPS hacia red de servidores
 permit tcp 10.255.1.0 0.0.0.255 eq www 10.0.1.0 0.255.254.255 gt 1023 ! HTTP hacia red de servidores
 permit udp host 10.255.1.10 eq domain 10.0.1.0 0.255.254.255 gt 1023 ! DNS desde servidor web
 deny ip any 10.0.0.0 0.255.255.255             ! Bloquear acceso a red corporativa
 permit tcp host 10.255.1.10 eq 443 any             ! Permitir HTTPS de salida del servidor
 permit tcp 10.255.1.0 0.0.0.255 any eq 443             ! Permitir HTTPS desde clientes
 permit icmp 10.255.1.0 0.0.0.255 any               ! Permitir ping

ip access-list extended CONTROL-VLAN20                  ! ACL para restringir tráfico de la VLAN20
```

```
 permit ospf any any                                ! Permitir OSPF
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985      ! Permitir HSRP
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985    ! Permitir HSRP v2
 permit tcp 10.255.2.0 0.0.0.255 eq 443 10.0.0.0 0.255.254.255 gt 1023  ! HTTPS hacia red de servidores
 permit tcp 10.255.2.0 0.0.0.255 eq www 10.0.0.0 0.255.254.255 gt 1023 ! HTTP hacia red de servidores
 permit tcp host 10.255.2.10 eq pop3 10.0.0.0 0.255.254.255 gt 1023    ! POP3 desde servidor correo
 permit tcp host 10.255.2.10 eq smtp 10.0.0.0 0.255.254.255 gt 1023    ! SMTP desde servidor correo
 permit udp host 10.255.2.10 eq domain 10.0.0.0 0.255.254.255 gt 1023  ! DNS desde servidor correo
 deny ip any 10.0.0.0 0.255.255.255                 ! Bloquear acceso a red corporativa
 permit tcp host 10.255.2.10 eq 443 any               ! HTTPS desde servidor
 permit tcp 10.255.2.0 0.0.0.255 any eq 443             ! HTTPS desde clientes
 permit icmp 10.255.2.0 0.0.0.255 any                 ! Permitir ping


!Aplicación de ACLs en interfaces VLAN
interface Vlan10
 ip access-group CONTROL-VLAN10 in                  ! Aplicar ACL a VLAN10

interface Vlan20
 ip access-group CONTROL-VLAN20 in
```

3-SWITCHES DE DISTRIBUCIÓN de REDES INTERNAS (DL-SW11, DL-SW12, DL-SW21, DL-SW22):

DL-SW11:

! ACL para VLANs impares (10 y 30)

ip access-list extended CONTROL-VLANS-IMPARES

NO FILTRA TRÁFICO POR Ips IMPARES -> ESO SE INDICA EN LA ASIGNACIÓN DE INTERFACES (VLAN 10 Y 30)

permit ospf any any ! Permitir OSPF para enrutamiento interno

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985 ! Permitir CDP u otros protocolos multicast

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985 ! Permitir protocolos de red internos

permit tcp 10.0.1.0 0.255.254.255 10.255.1.0 0.0.0.255 eq www ! HTTP a servidores web internos

permit tcp 10.0.1.0 0.255.254.255 10.255.1.0 0.0.0.255 eq 443 ! HTTPS a servidores web internos

permit udp 10.0.1.0 0.255.254.255 host 10.255.1.10 eq domain ! DNS a servidor DNS interno

deny ip any 10.0.0.0 0.255.255.255 ! Bloquear acceso a la red interna desde fuera

permit tcp 10.0.1.0 0.255.254.255 any eq 80 ! Permitir navegación HTTP a Internet

permit tcp 10.0.1.0 0.255.254.255 any eq 443 ! Permitir navegación HTTPS a Internet

permit icmp 10.0.1.0 0.255.254.255 any ! Permitir ping hacia fuera

**mascara 255.254.255.255: abarca 10.0.0.0 - 10.1.255.255QUE

!ACL para VLANs pares (20 y 40)

ip access-list extended CONTROL-VLANS-PARES

permit ospf any any

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985

```
 permit tcp 10.0.0.0 0.255.254.255 10.255.2.0 0.0.0.255 eq www   ! HTTP a servidor web VLAN pares
 permit tcp 10.0.0.0 0.255.254.255 10.255.2.0 0.0.0.255 eq 443   ! HTTPS a servidor web VLAN pares
 permit tcp 10.0.0.0 0.255.254.255 host 10.255.2.10 eq smtp      ! Enviar correo
 permit tcp 10.0.0.0 0.255.254.255 host 10.255.2.10 eq pop3      ! Leer correo
 permit udp 10.0.0.0 0.255.254.255 host 10.255.2.10 eq domain    ! DNS
 deny ip any 10.0.0.0 0.255.255.255
 permit tcp 10.0.0.0 0.255.254.255 any eq 80
 permit tcp 10.0.0.0 0.255.254.255 any eq 443
 permit icmp 10.0.0.0 0.255.254.255 any

! Aplicar ACLs en interfaces VLAN
interface Vlan10
 ip access-group CONTROL-VLANS-IMPARES in      ! VLAN 10 es impar
!
interface Vlan20
 ip access-group CONTROL-VLANS-PARES in        ! VLAN 20 es par
!
interface Vlan30
 ip access-group CONTROL-VLANS-IMPARES in      ! VLAN 30 es impar
!
interface Vlan40
 ip access-group CONTROL-VLANS-PARES in        ! VLAN 40 es par
```

DL-SW12:

```
!ACL para VLANs impares (10 y 30)
```

```
ip access-list extended CONTROL-VLANS-IMPARES
 permit ospf any any                    ! Permitir OSPF para enrutamiento interno
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985     ! Permitir CDP u otros protocolos multicast
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985   ! Permitir protocolos de red internos
 permit tcp 10.0.1.0 0.255.254.255 10.255.1.0 0.0.0.255 eq www   ! HTTP a servidores web internos
 permit tcp 10.0.1.0 0.255.254.255 10.255.1.0 0.0.0.255 eq 443   ! HTTPS a servidores web internos
 permit udp 10.0.1.0 0.255.254.255 host 10.255.1.10 eq domain    ! DNS a servidor DNS interno
 deny ip any 10.0.0.0 0.255.255.255           ! Bloquear acceso a la red interna desde fuera
 permit tcp 10.0.1.0 0.255.254.255 any eq 443   ! Permitir navegación HTTPS a Internet
 permit icmp 10.0.1.0 0.255.254.255 any         ! Permitir ping hacia fuera



!  ACL para VLANs pares (20 y 40)
ip access-list extended CONTROL-VLANS-PARES
 permit ospf any any
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985
 permit tcp 10.0.0.0 0.255.254.255 10.255.2.0 0.0.0.255 eq www   ! HTTP a servidor web VLAN pares
 permit tcp 10.0.0.0 0.255.254.255 10.255.2.0 0.0.0.255 eq 443   ! HTTPS a servidor web VLAN pares
 permit tcp 10.0.0.0 0.255.254.255 host 10.255.2.10 eq smtp      ! Enviar correo
 permit tcp 10.0.0.0 0.255.254.255 host 10.255.2.10 eq pop3      ! Leer correo
 permit udp 10.0.0.0 0.255.254.255 host 10.255.2.10 eq domain    ! DNS
 deny ip any 10.0.0.0 0.255.255.255
 permit tcp 10.0.0.0 0.255.254.255 any eq 443
 permit icmp 10.0.0.0 0.255.254.255 any
```

!Aplicar ACLs en interfaces VLAN

interface Vlan10

ip access-group CONTROL-VLANS-IMPARES in ! VLAN 10 es impar

interface Vlan20

ip access-group CONTROL-VLANS-PARES in ! VLAN 20 es par

interface Vlan30

ip access-group CONTROL-VLANS-IMPARES in ! VLAN 30 es impar

interface Vlan40

ip access-group CONTROL-VLANS-PARES in ! VLAN 40 es par

DL-SW21:

! ACL para VLANs impares (10 y 30)

ip access-list extended CONTROL-VLANS-IMPARES

permit ospf any any ! Permitir OSPF para enrutamiento interno

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985 ! Permitir HSRP

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985 ! Permitir HSRP v2

permit tcp 10.2.1.0 0.0.0.255 10.255.1.0 0.0.0.255 eq www ! HTTP a servidores web internos

permit tcp 10.2.1.0 0.0.0.255 10.255.1.0 0.0.0.255 eq 443 ! HTTPS a servidores web internos

permit udp 10.2.1.0 0.0.0.255 host 10.255.1.10 eq domain ! DNS a servidor DNS interno

deny ip any 10.0.0.0 0.255.255.255 ! Bloquear acceso a la red interna

```
 permit tcp 10.2.1.0 0.0.0.255 any eq 443                ! Permitir navegación HTTPS a Internet
 permit tcp 10.2.1.0 0.0.0.255 any eq www                ! Permitir navegación HTTP a Internet
 permit icmp 10.2.1.0 0.0.0.255 any                      ! Permitir ping hacia fuera

! ACL para VLANs pares (20 y 40)
ip access-list extended CONTROL-VLANS-PARES
 permit ospf any any
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985
 permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985
 permit tcp 10.2.2.0 0.0.0.255 10.255.2.0 0.0.0.255 eq www    ! HTTP a servidor web VLAN pares
 permit tcp 10.2.2.0 0.0.0.255 10.255.2.0 0.0.0.255 eq 443    ! HTTPS a servidor web VLAN pares
 permit tcp 10.2.2.0 0.0.0.255 host 10.255.2.10 eq smtp       ! Enviar correo
 permit tcp 10.2.2.0 0.0.0.255 host 10.255.2.10 eq pop3       ! Leer correo
 permit udp 10.2.2.0 0.0.0.255 host 10.255.2.10 eq domain     ! DNS
 deny ip any 10.0.0.0 0.255.255.255
 permit tcp 10.2.2.0 0.0.0.255 any eq 443                ! HTTPS a Internet
 permit tcp 10.2.2.0 0.0.0.255 any eq www                ! HTTP a Internet
 permit icmp 10.2.2.0 0.0.0.255 any                      ! Ping

! Aplicar ACLs en interfaces VLAN
interface Vlan10
 ip access-group CONTROL-VLANS-IMPARES in      ! VLAN 10 es impar

interface Vlan20
 ip access-group CONTROL-VLANS-PARES in        ! VLAN 20 es par

interface Vlan30
 ip access-group CONTROL-VLANS-IMPARES in      ! VLAN 30 es impar
```

interface Vlan40

ip access-group CONTROL-VLANS-PARES in ! VLAN 40 es par

DL-SW22:

! ACL para VLANs impares (10 y 30)

ip access-list extended CONTROL-VLANS-IMPARES

permit ospf any any ! Permitir OSPF para enrutamiento interno

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985 ! Permitir HSRP

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985 ! Permitir HSRP v2

permit tcp 10.2.3.0 0.0.0.255 10.255.1.0 0.0.0.255 eq www ! HTTP a servidores web internos

permit tcp 10.2.3.0 0.0.0.255 10.255.1.0 0.0.0.255 eq 443 ! HTTPS a servidores web internos

permit udp 10.2.3.0 0.0.0.255 host 10.255.1.10 eq domain ! DNS a servidor DNS interno

deny ip any 10.0.0.0 0.255.255.255 ! Bloquear acceso a la red interna

permit tcp 10.2.3.0 0.0.0.255 any eq 443 ! Permitir navegación HTTPS a Internet

permit tcp 10.2.3.0 0.0.0.255 any eq www ! Permitir navegación HTTP a Internet

permit icmp 10.2.3.0 0.0.0.255 any ! Permitir ping hacia fuera

! ACL para VLANs pares (20 y 40)

ip access-list extended CONTROL-VLANS-PARES

permit ospf any any

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.2 eq 1985

permit udp 10.0.0.0 0.255.255.255 host 224.0.0.102 eq 1985

permit tcp 10.2.4.0 0.0.0.255 10.255.2.0 0.0.0.255 eq www ! HTTP a servidor web VLAN pares

```
 permit tcp 10.2.4.0 0.0.0.255 10.255.2.0 0.0.0.255 eq 443    ! HTTPS a servidor web VLAN pares
 permit tcp 10.2.4.0 0.0.0.255 host 10.255.2.10 eq smtp       ! Enviar correo
 permit tcp 10.2.4.0 0.0.0.255 host 10.255.2.10 eq pop3       ! Leer correo
 permit udp 10.2.4.0 0.0.0.255 host 10.255.2.10 eq domain     ! DNS
 deny ip any 10.0.0.0 0.255.255.255
 permit tcp 10.2.4.0 0.0.0.255 any eq 443                     ! HTTPS a Internet
 permit tcp 10.2.4.0 0.0.0.255 any eq www                     ! HTTP a Internet
 permit icmp 10.2.4.0 0.0.0.255 any                           ! Ping

! Aplicar ACLs en interfaces VLAN
interface Vlan10
 ip access-group CONTROL-VLANS-IMPARES in      ! VLAN 10 es impar

interface Vlan20
 ip access-group CONTROL-VLANS-PARES in        ! VLAN 20 es par

interface Vlan30
 ip access-group CONTROL-VLANS-IMPARES in      ! VLAN 30 es impar

interface Vlan40
 ip access-group CONTROL-VLANS-PARES in        ! VLAN 40 es par
```

Comprobaciones del funcionamiento:

1.Verificaciones desde subredes 10.1.x.x y 10.2.x.x hacia servidores (10.255.x.x) y comprobación de tráfico no existente entre subredes.

Ejemplos:

1.1-Desde 10.1.1.0/24:

- HTTP al servidor 10.255.1.10

telnet 10.255.1.10 80

Trying 10.255.1.10 ...Open

- DNS (UDP) al servidor 10.255.1.10

nslookup www.google.com 10.255.1.10

Server: [10.255.1.10]

Address: 10.255.1.10

Non-authoritative answer:

Name: www.google.com

Address: 8.8.8.8

- No hay tráfico entre subredes

C:\>ping 10.1.2.10

Pinging 10.1.2.10 with 32 bytes of data:

Reply from 10.1.1.253: Destination host unreachable.

Reply from 10.1.1.253: Destination host unreachable.

Reply from 10.1.1.253: Destination host unreachable.

Reply from 10.1.1.253: Destination host unreachable.

Ping statistics for 10.1.2.10:

Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 10.2.1.10

Pinging 10.2.1.10 with 32 bytes of data:

Reply from 10.1.1.253: Destination host unreachable.
Reply from 10.1.1.253: Destination host unreachable.
Reply from 10.1.1.253: Destination host unreachable.
Reply from 10.1.1.253: Destination host unreachable.

Ping statistics for 10.2.1.10:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

1.2-Desde 10.2.1.0/24:

- HTTP al servidor 10.255.2.10

telnet 10.255.2.10 80
Trying 10.255.2.10 ...Open

- DNS (UDP) al servidor 10.255.2.10

nslookup www.google.com 10.255.2.10

Server: [10.255.2.10]
Address: 10.255.2.10

Non-authoritative answer:
Name: www.google.com
Address: 8.8.8.8

- SMTP Y POP3

telnet 10.255.2.10 25 ← SMTP

telnet 10.255.2.10 110 ← POP3

2.Acceso a Internet (HTTP, HTTPS e ICMP) desde las subredes privadas 10.1.x.x y 10.2.x.x.

1.1-Desde 10.1.1.0/24:

telnet 8.8.8.8 80 !verificar HTTP

Trying 8.8.8.8 ...Open

telnet 8.8.8.8 443 !verificar HTTPS

Trying 8.8.8.8 ...Open

ping 8.8.8.8 !verificar ICMP

Pinging 8.8.8.8 with 32 bytes of data:

Reply from 8.8.8.8: bytes=32 time=10ms TTL=122

Reply from 8.8.8.8: bytes=32 time=4ms TTL=122

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Ping statistics for 8.8.8.8:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 2ms, Maximum = 10ms, Average = 4ms

1.1-Desde 10.2.1.0/24:

telnet 8.8.8.8 80 !verificar HTTP

Trying 8.8.8.8 ...Open

telnet 8.8.8.8 443 !verificar HTTPS

Trying 8.8.8.8 ...Open

ping 8.8.8.8 !verificar ICMP

Pinging 8.8.8.8 with 32 bytes of data:

Reply from 8.8.8.8: bytes=32 time=10ms TTL=122

Reply from 8.8.8.8: bytes=32 time=4ms TTL=122

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Ping statistics for 8.8.8.8:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 2ms, Maximum = 10ms, Average = 4ms

3.Verificar bloqueo de tráfico no permitido

- Desde 10.1.1.x a 10.255.2.x → **debe estar bloqueado**

telnet 10.255.2.10 80

Trying 10.255.2.10 ...

% Connection timed out; remote host not responding

- Desde 10.2.1.x a 10.255.1.x → **debe estar bloqueado**

telnet 10.255.1.10 80

Trying 10.255.1.10 ...

% Connection timed out; remote host not responding

4. Comprobación de tráfico de retorno (CBAC)

-Desde 10.1.1.10/24:

ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Reply from 8.8.8.8: bytes=32 time=2ms TTL=122

Ping statistics for 8.8.8.8:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 2ms, Maximum = 2ms, Average = 2ms

Si vemos respuestas (reply from...), entonces ICMP está funcionando y las respuestas (eco-reply) están volviendo correctamente, lo que confirma el tráfico de retorno para ICMP.

Si mantenemos el ping con -t y vamos al firewall, podemos comprobar si funciona las CBACS:

show ip inspect sessions:

- Muestra las sesiones activas que están siendo monitoreadas por **CBAC**.
- En este caso, tienes tres sesiones:
  - Una ICMP de **192.0.0.10 a 8.8.8.8**.
  - Dos sesiones **TCP** entre **192.0.0.10 y 192.0.0.13**.
- Los estados SIS_OPEN y SIS_OPENING indican que las sesiones están activas, pero algunas aún no han sido completamente abiertas o cerradas.

show ip inspect statistics:

- Proporciona estadísticas sobre las sesiones y el estado de la inspección de paquetes:
    - **3 sesiones activas**.
    - El proceso ha manejado **5076 paquetes**.
    - Las **sesiones máximas** fueron de 4 activas, pero actualmente hay 3.
    - **TCP reassembly** está vacío, lo que significa que no se han recibido paquetes fuera de orden que requieran reensamblaje.

show running-config | section inspect:

- Muestra la configuración de inspección de tráfico:
    - **RETURN-TRAFFIC** está configurado para inspeccionar el tráfico **TCP** (con un timeout de 3600 segundos) e **ICMP** (con un timeout de 10 segundos).
    - La directiva ip inspect RETURN-TRAFFIC out indica que la inspección está habilitada para el tráfico de salida.

5.Verificación de Port Forwarding

telnet 192.0.0.13 443 ← Debe redirigir a 10.255.1.10:443

telnet 192.0.0.14 443 ← Debe redirigir a 10.255.2.10:443

COMANDOS:

1-Configuraciones ACL Estándar

<u>-Numeradas:</u>

- access-list [num ACL (1-99) (1300-1999)] {permit|deny} descripción [dirección-origen] [wildcard]: crea una regla que permite o deniega tráfico desde una IP o red específica usando máscara wildcard
  *equivalente a: access-list [num ACL (1-99)] {permit|deny} host ip (sin máscara wildcard)

- interface [tipo-interfaz] [número]: entra al modo de configuración de una interfaz específica del router

- ip access-group [num ACL] {in|out}: aplica una ACL a la interfaz seleccionada, indicando si filtra el tráfico que entra (in) o que sale (out)

<u>-Nombradas:</u>

- ip access-list standard [nombre]: crea ACL estándar nombrada
  *nombre: puede incluir letras y números, pero no signos de puntuación o espacios. Se recomienda la escritura en mayúsculas.

- {permit|deny} descripcion [dirección-origen] [wildcard]: agrega reglas dentro del modo ACL

- interface [tipo-interfaz] [número]

- ip access-group [nombre] {in|out}: aplica la ACL nombrada

2-Configuraciones de ACL Extendida

<u>-Numeradas:</u>

- access-list [num ACL (100-199) (2000-2699)] {permit|deny} descripcion [protocolo] [origen] [wildcard-origen] [destino] [wildcard-destino] [operador] [puerto]: regla extendida numerada

*puertos comunes:

HTTP – 80 || HTTPS – 443

FTP – 20 (datos) | 21 (control)

SSH – 22 || Telnet – 23

SMTP - 25 || POP3 - 110

DNS – 53

HSRP - 1985 (udp)

*established: Permite tráfico que vuelve a la LAN, pero solo si es parte de conexiones ya iniciadas desde dentro.

El parámetro established verifica que el tráfico tiene el bit ACK o RST, lo que indica que es respuesta de una conexión previa.

- interface [tipo-interfaz] [número]
- ip access-group [num-ACL] {in|out}: aplica la ACL a la interfaz

-Nombradas:

- ip access-list extended [NOMBRE]: crea ACL extendida nombrada
- permit|deny [protocolo] [IP origen] [wildcard] [IP destino] [wildcard] [puerto]: regla en modo ACL
- interface [tipo interfaz] [numero]
- ip access-group [NOMBRE] {in|out}: aplica la ACL nombrada

BORRAR UNA ACL:

1-Eiminar la ACL de la interfaz (aplicación de la ACL)

no ip [interfaz] [num ACL] in

2-Borrar la configuración de la ACL:

no access-list [num ACL]

3-Modificación de ACLs:

1-Editor de texto:

Mostrar la configuración actual de una ACL
show running-config | include access-list [número]

Copiar la configuración a un editor de texto externo

Eliminar completamente una ACL existente
no access-list [número]

Recrear o modificar la ACL línea por línea con los cambios deseados
access-list [número] {permit|deny} [criterios]

Verificar los cambios realizados
show running-config | include access-list [número]

2-Números de secuencia: Usando editor de texto externo

Reemplazar una entrada existente:

• Ver los números de secuencia actuales:
show access-lists [número]

• Entrar en modo de edición de la ACL:
ip access-list {standard|extended} [número|nombre]

• Eliminar una entrada específica:
no [número-secuencia]

• Añadir una nueva entrada con el mismo número de secuencia (modificada):
[número-secuencia] {permit|deny} [criterios]

Añadir una nueva entrada (sin borrar otras):

• Entrar en modo de edición de la ACL:
ip access-list {standard|extended} [número|nombre]

• Insertar nueva instrucción entre otras existentes:
[número-secuencia] {permit|deny} [criterios]

• Verificar los cambios:
show access-lists

show access-lists [número]

3-Verificación de ACLS:

- show access-lists: resumen de todas las ACLs. Diferencia entre Standard (estándar) y extended (extendidas)
- show access-lists [num ACL]: muestra de que tipo es la ACL y su configuración con sus números de secuencia por orden.
- show ip access-lists: ver tipo de ACL, reglas y aplicación en interfaz.
- show ip interface: ver si la ACL está aplicada a una interfaz y en qué dirección (in/out)
- show running-config: Ver toda la configuración del router (incluye ACLs aplicadas)

4-Estadísticas de ACLs:

Ver cuántos paquetes coinciden con cada regla de la ACL:

- show access-lists
- show access-lists [número]

*matches: paquetes denegados o permitidos por una instrucción.

5-CBAC (reglas de Inspección):

Dentro de una interfaz a la que le aplicamos una ACL, se utiliza:

1-Crear la regla de inspección:

- ip inspect name inspection_name protocol [alert] {on | off}] [audit-trail {on | off}] [timeout seconds]

* alert {on | off}: Activa o desactiva las alertas para eventos de inspección.

-on: Envía alertas cuando se detectan eventos relevantes.

-off: No envía alertas.

* audit-trail {on | off}: Activa o desactiva el registro de auditoría de eventos de inspección.

-on: Registra información detallada sobre las inspecciones.

-off: No registra información de auditoría.

*timeout: tiempo que dura la sesión si no hay actividad

2-Aplicar la regla de inspección a una interfaz

- interface INTERFAZ
- ip inspect NOMBRE_REGLA in

3-Eliminar una regla de inspección:

- no ip inspect

4-Troubleshooting CBAC:

Es el proceso para revisar si las CBAC están funcionando correctamente, si están creadas las reglas temporales y si están permitiendo el tráfico esperado.

- show ip inspect [opciones]: permite ver que reglas de inspección están activas, que sesiones están abiertas y que ACLs temporales se han creado automáticamente.

Ejemplos de opciones:

- show ip inspect config: muestra la configuración actual de inspección
- show ip inspect sessions: muestra las conexiones activas que están siendo inspeccionadas
- show ip inspect all: muestra toda la configuración.

*debug ip inspect: habilita la depuración de las funciones de inspección de CBAC en dispositivos Cisco.

6-Configuración de NAT:

1-Configuración de NAT estático (sin sobrecarga) -> STATIC

- ip nat inside source **static** <ip-local> <ip-global>: mapea una IP privada a una IP pública.
- ip nat {inside|outside}

2-Configuración de NAT dinámico (sin sobrecarga) -> LIST

Utiliza un grupo de IPs públicas para asignar dinámicamente a las IPs privadas:

Paso1: definir el conjunto de direcciones públicas:

- ip nat pool <nombre> <ip-inicial> <ip-final> netmask <máscara>

Paso2: definir que IPs privadas pueden ser traducidas:

- access-list <número> permit <origen> <wildcard>

Paso3: relacionar las IPs privadas con las públicas:

- ip nat inside source **list** <número ACL> pool <nombre>

3-Configuración de NAT Sobrecargado (PAT) -> OVERLOAD (AL FINAL)

Permite que múltiples IPs privadas compartan una sola IP pública utilizando diferentes números de puerto.

- ip nat inside source list <número ACL> interface <tipo-número> overload

3-Configuración de Port Mapping:

El Port Mapping permite que un puerto específico de una IP pública se asocie con un puerto de una IP privada. Esto es útil para acceder a servicios específicos en una red privada desde el exterior.

1.Relacionar la IP y puerto privado con la IP y puerto público:

- ip nat inside source static [tcp|udp] <ip-local-interna> <puerto-local-interno> <ip-global-interna> <puerto-global-interno>

2. Identificar la interfaz interna:

- interface <tipo-int nº-int>
- ip nat inside

3. Identificar la interfaz externa:

- interface <tipo-int nº-int>
- ip nat outside

DESACTIVACIÓN DE TRADUCCIONES:

4-Comandos de diagnóstico:

- show ip nat translations: Muestra la tabla de traducciones NAT actuales.
- show ip nat statistics: Proporciona estadísticas sobre el funcionamiento de NAT.
- debug ip nat: Habilita la depuración de las funciones de NAT en tiempo real.

Activar security plus:

Show license all

➔ Securityk9

License boot module c2900 tecnhology-package securityk9

Wr

Reload

EJEMPLO EXAMEN:

Parte 1: ACLs

1. ¿Qué tipo de ACL se utiliza para filtrar tráfico según direcciones IP y permitir o denegar acceso a una red específica?
    - a) ACL estándar
    - b) ACL extendida
    - c) ACL dinámica
    - d) ACL implícita

a)

2. ¿Cuál es la principal diferencia entre una ACL estándar y una extendida?
    - a) La ACL estándar solo filtra por direcciones IP, mientras que la extendida permite filtrar por puertos, protocolos, y direcciones.
    - b) La ACL estándar permite filtrado por puertos, mientras que la extendida solo filtra por direcciones IP.
    - c) La ACL estándar filtra el tráfico basado en tipos de tráfico, mientras que la extendida filtra solo por direcciones IP.
    - d) No hay diferencia, ambas son exactamente iguales.

a)

3. ¿Qué comando utilizarías para aplicar una ACL extendida llamada DENY_ACL a la interfaz FastEthernet0/0?
    - a) ip access-group DENY_ACL in
    - b) access-list DENY_ACL apply
    - c) ip acl apply DENY_ACL
    - d) access-group DENY_ACL

a)

4. ¿Qué número de ACL estándar permitiría el tráfico de solo la dirección IP 192.168.1.1?
    - a) 1
    - b) 50
    - c) 100
    - d) 10

a)

5. ¿Cuál es la sintaxis correcta para negar todo el tráfico desde cualquier red en una ACL extendida?
    - a) deny ip any any
    - b) deny all any any
    - c) deny ip 0.0.0.0 0.0.0.0
    - d) deny all

a)

Parte 2: CBAC (Context-Based Access Control)

6. ¿Cuál es la función principal de CBAC en un router o firewall?
    - a) Filtrar el tráfico entrante basado en ACLs.
    - b) Inspeccionar el tráfico y abrir dinámicamente las puertas de retorno para las sesiones establecidas.
    - c) Crear NAT estático para redes privadas.
    - d) Permitir tráfico sin inspección.

b)

7. ¿Qué comando se utiliza para habilitar la inspección de tráfico TCP en una interfaz de salida?
    - a) ip inspect name INSPECT_TCP out
    - b) ip inspect name INSPECT_TCP in
    - c) ip inspect tcp out
    - d) ip inspect INSPECT_TCP

a)

8. ¿Cómo se configura un tiempo de espera para las sesiones TCP inspeccionadas en CBAC?
    - a) ip inspect name TIMEOUT tcp timeout 300
    - b) ip inspect name TIMEOUT tcp session-time 300
    - c) ip inspect name TIMEOUT timeout 300
    - d) ip inspect name TIMEOUT tcp timeout 3600

a)

9. ¿Cuál es la utilidad del comando show ip inspect sessions?
    - a) Mostrar las estadísticas de tráfico de la red.
    - b) Ver las sesiones de inspección activas y sus detalles.
    - c) Mostrar el estado de las ACLs configuradas.
    - d) Ver las interfaces configuradas para NAT.

b)

10. ¿Qué ocurre si no se configura un comando de inspección para el tráfico saliente en CBAC?
    - a) El tráfico saliente se bloquea por defecto.
    - b) El tráfico saliente no es inspeccionado y las respuestas a sesiones iniciadas desde la red interna pueden ser bloqueadas.
    - c) Todo el tráfico saliente es permitido sin inspección.
    - d) El tráfico saliente es auditado pero no inspeccionado.

b)

Parte 3: NAT (Network Address Translation)

11. ¿Qué tipo de NAT se utiliza cuando un solo dispositivo en una red privada comparte una única dirección IP pública?
    - a) NAT Estático
    - b) NAT Dinámico
    - c) NAT con sobrecarga (PAT)
    - d) NAT Manual

a)

12. ¿Cuál es la principal diferencia entre NAT estático y NAT dinámico?
    - a) En NAT estático, se mapea una dirección IP privada a una pública, mientras que en NAT dinámico, se asigna una IP pública de un pool a una IP privada.
    - b) NAT estático es solo para redes grandes, y NAT dinámico solo para redes pequeñas.
    - c) En NAT dinámico, se mapea una dirección privada a una pública permanentemente.
    - d) No hay diferencia, ambos se utilizan para el mismo propósito.

a)

13. ¿Qué comando se utiliza para configurar NAT con sobrecarga (PAT)?
    - a) ip nat inside source list 1 interface FastEthernet0 overload
    - b) ip nat inside source list 1 pool NAT_POOL overload
    - c) ip nat inside source static 192.168.1.10 203.0.113.10
    - d) ip nat outside source list 1 interface FastEthernet0 overload

a)

14. ¿Cuál es el propósito del comando ip nat inside y ip nat outside?
    - a) Identificar las interfaces internas y externas para que el tráfico pueda ser traducido.
    - b) Configurar NAT de sobrecarga para las interfaces internas.
    - c) Configurar las ACLs para los accesos internos y externos.
    - d) Configurar la red privada y la pública.

a)

15. ¿Qué comando se usaría para verificar la tabla de NAT en un router?
    - a) show ip nat translations
    - b) show nat stats
    - c) show ip route nat
    - d) show nat

a)

Parte 4: Preguntas Variadas

16. ¿Cuál de los siguientes es un beneficio de utilizar CBAC en lugar de una ACL estática?
    - a) La ACL estática permite abrir puertos automáticamente.
    - b) CBAC puede permitir sesiones de retorno dinámicamente, mientras que las ACLs no lo hacen.
    - c) Las ACLs filtran de forma más eficiente que CBAC.
    - d) CBAC es más fácil de configurar que las ACLs.

b)

17. ¿Qué comando te permite visualizar las estadísticas de la inspección de tráfico de CBAC, incluyendo las sesiones abiertas?
    - a) show ip inspect statistics
    - b) show ip inspect sessions
    - c) show ip nat translations
    - d) show ip access-lists

b)

18. ¿Qué puede suceder si no configuras una ACL en un router que está haciendo NAT?
    - a) No habrá traducción de direcciones y todo el tráfico será denegado.
    - b) Todo el tráfico será permitido sin ningún control.
    - c) El tráfico de salida se verá bloqueado si no está autorizado.
    - d) El router no podrá hacer NAT correctamente.

c)

19. ¿Qué comando se utiliza para eliminar una ACL previamente configurada?
    - a) no access-list 100
    - b) access-list 100 delete
    - c) ip access-group 100 remove
    - d) clear ip access-list 100

a)

20. En un entorno de NAT con sobrecarga, ¿qué tipo de dirección IP pública se utilizaría?
    - a) Una dirección IP pública fija asignada.
    - b) Un rango de direcciones IP públicas.
    - c) Una dirección IP pública asignada dinámicamente desde un pool.
    - d) Una dirección IP privada asignada manualmente.

c)